On-line construction of a small automaton for a finite set of words

Maxime Chrochemore and Laura Giambruno

Institut Gaspard-Monge, Université Paris-Est
Dipartimento di Matematica e Applicazioni, Università di Palermo, Palermo

August 31,2009

Main focus

Design a “light” algorithm for the on-line construction of a small automaton recognising a finite set of words in linear time.

- Finite sets of words X on a finite alphabet A.
- the *length* n of X is the sum of the lengths of the words in X:

$$n = \sum_{i=1}^{m} |u_i|$$

Main focus

Design a "light" algorithm for the on-line construction of a small automaton recognising a finite set of words in linear time.

- Finite sets of words X on a finite alphabet A.
- the *length* n of X is the sum of the lengths of the words in X:

$$n = \sum_{i=1}^{m} |u_i|$$

Motivations

- Interesting for parsing natural text and for motif detection
- Used in many software like the intensively used BLAST
- Dictionaries used for natural languages can contain a large number of words.

Automata for finite sets of words: classical construction

- represent a list X by a trie
- minimise the trie to get the minimal automaton of the finite set of words of the list.

This solution requires a large memory space to store the temporary large data structure.

Automata for finite sets of words: classical construction

- represent a list X by a trie
- minimise the trie to get the minimal automaton of the finite set of words of the list.

This solution requires a large memory space to store the temporary large data structure.

Some of other constructions

- pseudo-minimisation algorithm by Revuz (1991)
- algorithm that constructs a minimal automaton for an ordered set of strings by Daciuk et al. (2000)
- semi-incremental algorithm for constructing minimal acyclic deterministic automata by Watson (2003)
- efficient algorithm to insert a word in a minimal acyclic by Sgarbas et al. (2003)

What we propose

Intermediate solution

to build a rather small automaton with a light algorithm processing the list of words on- line in linear time on the length of the list.

- The aim is not to get the corresponding minimal automaton but just a small enough structure.
- However, the minimal automaton can be later obtained with Revuz linear algorithm (1992).

What we propose

Intermediate solution

to build a rather small automaton with a light algorithm processing the list of words on- line in linear time on the length of the list.

- The aim is not to get the corresponding minimal automaton but just a small enough structure.
- However, the minimal automaton can be later obtained with Revuz linear algorithm (1992).

What we propose

Intermediate solution

to build a rather small automaton with a light algorithm processing the list of words on- line in linear time on the length of the list.

- The aim is not to get the corresponding minimal automaton but just a small enough structure.
- However, the minimal automaton can be later obtained with Revuz linear algorithm (1992).

- the automaton can possibly be built on demand
- our solution avoids building a temporary large trie

Advantages of our algorithm

Simplicity, linear time algorithm, on-line construction and the fact that resulting automaton seems to be really close to minimal.

- the automaton can possibly be built on demand
- our solution avoids building a temporary large trie

Advantages of our algorithm

Simplicity, linear time algorithm, on-line construction and the fact that resulting automaton seems to be really close to minimal.

Definitions

Let A be a finite alphabet.
For $X = (x_0, \ldots, x_m)$ list of words, $|X|$ denotes the cardinality of X.

A *deterministic automaton* over A is $\mathcal{A} = (Q, i, T, \delta)$, where

- Q is a finite set of *states*
- i is the initial state
- $T \subseteq Q$ is the subset of *final* states
- $\delta : Q \times A \longrightarrow Q$ is the *transition function*

Definitions

Let $<$ be an order on the elements in A
Lexicographic order $<_{lex}$: for u, v in A^* we have that $u <_{lex} v$ if, and only if

- u is a prefix of v
- u and v have a prefix u_0 in common, $u = u_0 a u_1$, $v = u_0 b v_1$ and $a <_{lex} b$

Hypothesis

We consider a list of words X in A^* such that the list obtained reversing each word in X is sorted according to the lexicographic order.

Example

$X = (\texttt{aaa}, \texttt{ba}, \texttt{aab})$ satisfies our hypothesis.

Idea of the construction

- We define inductively a sequence of $|X| + 1$ automata $\mathcal{A}_X^0, \ldots, \mathcal{A}_X^{|X|}$.
- For each k, the automaton $\mathcal{A}_X^k$ recognises the language $\{x_0, \ldots, x_k\}$.
- In particular $\mathcal{A}_X^m$ will recognises X

For each k, there is a unique final state q_{fin} without any outgoing transitions.

Idea

Define $\mathcal{A}_X^0$
$\mathcal{A}_X^{k-1} \longrightarrow \mathcal{A}_X^k$ by adding a path in $\mathcal{A}_X^{k-1}$ in order to add x_k to $L(\mathcal{A}_X^{k-1})$.

Definitions

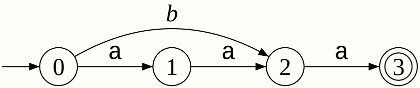
Let $\mathcal{A}_X^k$ with set of states Q_k and

$$H, Deg^-(j), PF : Q_k \longrightarrow N$$

such that for $j \in Q_k$:

- *Height*: $H(j)$ is the maximal length of paths from j to a final state.
- *Indegree*: $Deg^-(j)$ is the number of edges ending at j.
- *Paths toward final states*: for $j \neq q_{fin}$, $PF(j)$ is the number of paths starting at j and ending at final states and $PF(q_{fin}) = 1$.

Example



- $H(0) = 3$, $H(1) = 2$, $H(2) = 1$, $H(3) = 0$
- $Deg^-(0) = 0$, $Deg^-(1) = Deg^-(3) = 1$, $Deg^-(2) = 2$
- $PF(0) = 2$, $PF(1) = PF(2) = PF(3) = 1$

Definitions

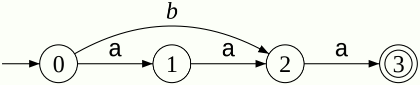
Let $\mathcal{A}_X^k$ with set of states Q_k and

$$H, Deg^-(j), PF : Q_k \longrightarrow N$$

such that for $j \in Q_k$:

- *Height*: $H(j)$ is the maximal length of paths from j to a final state.
- *Indegree*: $Deg^-(j)$ is the number of edges ending at j.
- *Paths toward final states*: for $j \neq q_{fin}$, $PF(j)$ is the number of paths starting at j and ending at final states and $PF(q_{fin}) = 1$.

Example



- $H(0) = 3, H(1) = 2, H(2) = 1, H(3) = 0$
- $Deg^-(0) = 0, Deg^-(1) = Deg^-(3) = 1, Deg^-(2) = 2$
- $PF(0) = 2, PF(1) = PF(2) = PF(3) = 1$

Definitions

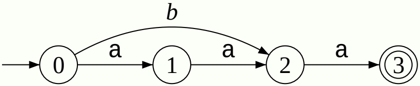
Let $\mathcal{A}_X^k$ with set of states Q_k and

$$H, Deg^-(j), PF : Q_k \longrightarrow N$$

such that for $j \in Q_k$:

- *Height*: $H(j)$ is the maximal length of paths from j to a final state.
- *Indegree*: $Deg^-(j)$ is the number of edges ending at j.
- *Paths toward final states*: for $j \neq q_{fin}$, $PF(j)$ is the number of paths starting at j and ending at final states and $PF(q_{fin}) = 1$.

Example



- $H(0) = 3, H(1) = 2, H(2) = 1, H(3) = 0$
- $Deg^-(0) = 0, Deg^-(1) = Deg^-(3) = 1, Deg^-(2) = 2$
- $PF(0) = 2, PF(1) = PF(2) = PF(3) = 1$

Construction of $\mathcal{A}_X^0$

Let $\mathcal{A}_X^0 = (Q_0, i_0, T_0, \delta_0)$ be a path with label x_0 from $i_0 = 0$ to $|x_0| = q_{fin}$ unique final state.

- The elements in Q_0 are integers
- $i_0 = 0$ and $T_0 = \{|x_0|\}$
- $L(\mathcal{A}_X^0) = \{x_0\}$

Example

$X = (\texttt{aaa}, \texttt{ba}, \texttt{aab})$

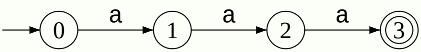


Figure: $\mathcal{A}_X^0$

Construction of $\mathcal{A}_X^k$ from $\mathcal{A}_X^{k-1}$

$$\mathcal{A}_X^{k-1} = (Q_{k-1}, i_{k-1}, T_{k-1}, \delta_{k-1}) \longrightarrow \mathcal{A}_X^k = (Q_k, i_k, T_k, \delta_k)$$

- $i_k = 0$
- $u \longrightarrow$ the longest prefix in common between x_k and the elements in $\{x_0, \ldots, x_{k-1}\}$.
- $s \longrightarrow$ the longest suffix in common between x_k and x_{k-1}.
- if s and u overlap we consider as s the suffix of x_k of length $|x_k| - |u| + 1$.
- $x_k = uws$, with $w \neq \varepsilon$.

For $X = (\mathtt{aaa}, \mathtt{ba}, \mathtt{aab})$ and for $x_2 = \mathtt{aab}$, u is $\mathtt{aa}$ and s is ε.

Construction of $\mathcal{A}_X^k$ from $\mathcal{A}_X^{k-1}$

$$\mathcal{A}_X^{k-1} = (Q_{k-1}, i_{k-1}, T_{k-1}, \delta_{k-1}) \longrightarrow \mathcal{A}_X^k = (Q_k, i_k, T_k, \delta_k)$$

- $i_k = 0$
- $u \longrightarrow$ the longest prefix in common between x_k and the elements in $\{x_0, \ldots, x_{k-1}\}$.
- $s \longrightarrow$ the longest suffix in common between x_k and x_{k-1}.
- if s and u overlap we consider as s the suffix of x_k of length $|x_k| - |u| + 1$.
- $x_k = uws$, with $w \neq \varepsilon$.

For $X = (\mathtt{aaa}, \mathtt{ba}, \mathtt{aab})$ and for $x_2 = \mathtt{aab}$, u is $\mathtt{aa}$ and s is ε.

Construction of $\mathcal{A}_X^k$ from $\mathcal{A}_X^{k-1}$

$p \longrightarrow$ the end state of the path in $\mathcal{A}_X^{k-1}$ starting at 0 with label u
$q \longrightarrow$ the state along the path from 0 with label x_{k-1} for which the sub-path from q to q_{fin} has label s

Example

$X = (\texttt{aaa}, \texttt{ba}, \texttt{aab})$

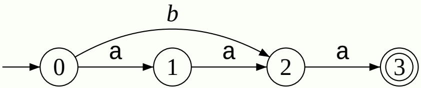


Figure: $\mathcal{A}_X^1$

For $x_2 = aab$, p is the state 2 and q is the state 3.

Construction of $\mathcal{A}_X^k$ from $\mathcal{A}_X^{k-1}$

General idea

The general idea of the construction of $\mathcal{A}_X^k$ from $\mathcal{A}_X^{k-1}$ would be to add a path from p to q with label w.

Example

$X = (\mathtt{aaa}, \mathtt{ba}, \mathtt{aab})$

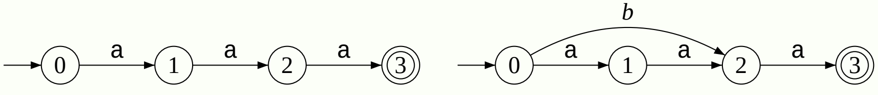


Figure: $\mathcal{A}_X^0$ and $\mathcal{A}_X^1$

The automaton $\mathcal{A}_X^1$ is obtained from $\mathcal{A}_X^0$ by adding the edge $(0, \mathsf{b}, 2)$.

Indegree control

Attention!

In general we cannot add a path from p to q with label w since we would add words other than x_k. We have to do some controls.

Example

$X = (\texttt{aaa}, \texttt{ba}, \texttt{aab})$

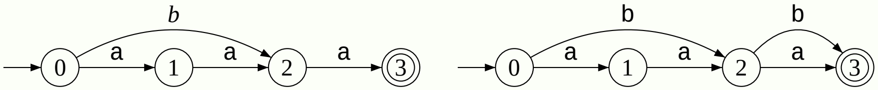


Figure: $\mathcal{A}_X^1$, the incorrect construction of $\mathcal{A}_X^2$

Since $Deg^-(2) > 1$, adding the edge $(2, \mathsf{b}, 3)$ leads to an automaton accepting $\{\texttt{aaa}, \texttt{ba}, \texttt{aab}, \texttt{bb}\}$.

Indegree control

Before adding a path from p to q, we have to do a transformation of the automaton $\mathcal{A}_X^1 \longrightarrow \mathcal{B}_X^1$.

Example

$X = (\texttt{aaa}, \texttt{ba}, \texttt{aab})$

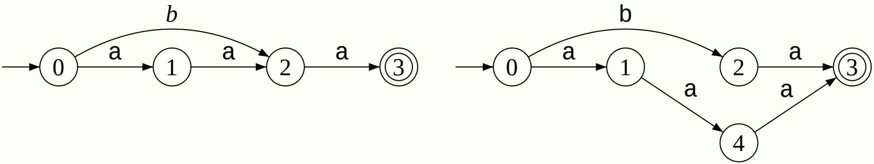


Figure: $\mathcal{A}_X^1$ and $\mathcal{B}_X^1$

$\mathcal{B}_X^1$ is obtained from $\mathcal{A}_X^1$ by doing a copy of the path from 0 to 4 with label $\texttt{aa}$.

Indegree control

Example

$X = (\texttt{aaa}, \texttt{ba}, \texttt{aab})$

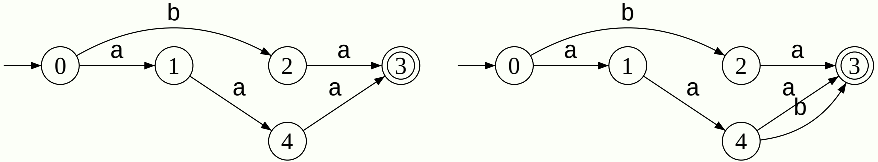


Figure: $\mathcal{B}_X^1$ and $\mathcal{A}_X^2$

$\mathcal{A}_X^2$ is obtained by adding the edge $(4, \mathsf{b}, 3)$.

Indegree control

If, in $\mathcal{A}_X^{k-1}$,in the path from 0 with label u there are states r with $Deg^-(r) > 1$ then

$$\mathcal{A}_X^{k-1} \longrightarrow \mathcal{B}_X^{k-1}$$

In this case:

- $\mathcal{B}_X^{k-1}$ is obtained by doing a copy of the path from r to p
- $\mathcal{B}_X^{k-1}$ is equivalent to $\mathcal{A}_X^{k-1}$
- $p \longrightarrow$ the end state of the path from 0 with label u in $\mathcal{B}_X^{k-1}$.

Prefix case

- If x_k is the prefix of a word in $\{x_0, \ldots, x_{k-1}\}$ then we add p to the set of final states.
- Otherwise we proceed with the following controls.

Path toward final states control

Example

$X = (\mathtt{aaa}, \mathtt{ba}, \mathtt{aab}, \mathtt{abb})$

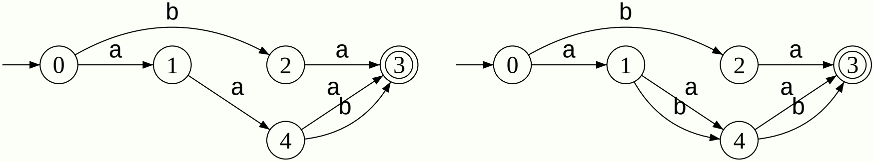


Figure: $\mathcal{A}_X^2$ and the incorrect construction of $\mathcal{A}_X^3$

We have $PF(4) = 2$.
Adding the edge $(1, \mathtt{b}, 4)$ to $\mathcal{A}_X^2$ leads to an automaton accepting $\{\mathtt{aaa}, \mathtt{ba}, \mathtt{aab}, \mathtt{abb}, \mathtt{aba}\}$.

Path toward final states control

Example

$X = (\mathtt{aaa}, \mathtt{ba}, \mathtt{aab}, \mathtt{abb})$

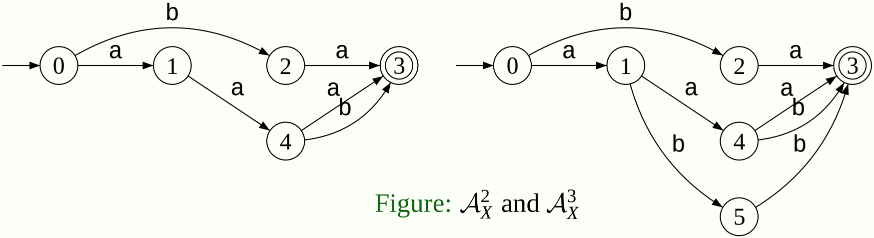


Figure: $\mathcal{A}_X^2$ and $\mathcal{A}_X^3$

$\mathcal{A}_X^3$ is obtained by adding the path from 1 to 3 with label $\mathtt{bb}$.
The state 3 is the first state q' in the path from $4 = q$ to 3 with $PF[q'] = 1$

Path toward final states control

If $PF(q) > 1$

- consider in the path from q to q_{fin} with label s the first state q' such that $PF[q'] = 1$, if it exists.
- redefine q as q'
- redefine w and s
- If there is no q' with $PF[q'] = 1$, redefine q as q_{fin} and w as ws.

Height control

Example

$X = (\texttt{aba}, \texttt{abbba})$

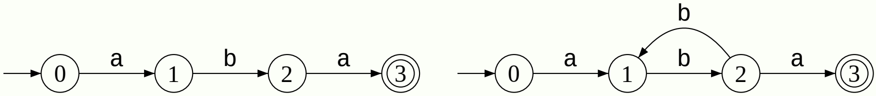


Figure: $\mathcal{A}_X^0$ and the incorrect construction of $\mathcal{A}_X^1$

We have that $p = 1 = q$ have the same H
Adding the edge $(2, \texttt{b}, 1)$ in $\mathcal{A}_X^0$ would lead to an automaton accepting the infinite language $\{\texttt{aba}, \texttt{a(bb)}^*\texttt{a}\}$.

Height control

Example

$X = (\texttt{aba}, \texttt{abbba})$

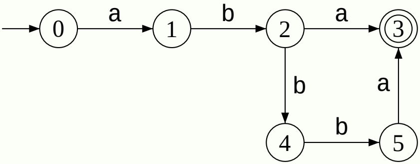


Figure: $\mathcal{A}_X^1$.

$\mathcal{A}_X^1$ is obtained by adding the path from 2 to 3 with label $\texttt{bba}$.
The state 3 is the first state q' in the path from 2 to 3 with $H[p] > H[q']$

Height control

If $H[p] \leq H[q]$

- consider in the path from q to q_{fin} with label s the first state q' such that $H[p] > H[q]$.
- redefine q as q'
- redefine w and s

Control on q_{fin}

Example

$X = (\texttt{aaa}, \texttt{ba}, \texttt{aab}, \texttt{abb}, \texttt{abbb})$

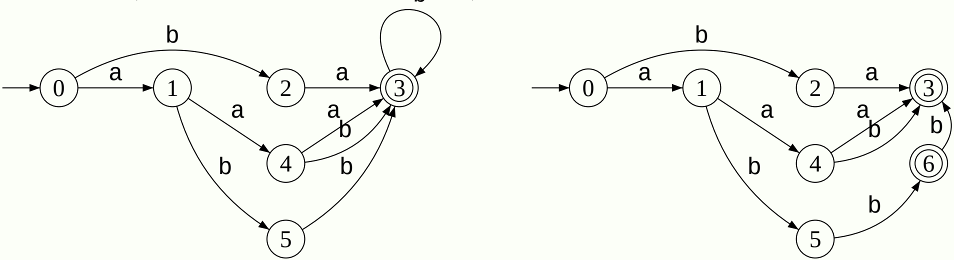


Figure: Incorrect construction of $\mathcal{A}_X^4$ and the right construction of $\mathcal{A}_X^4$

Adding an edge from $p = 3$ to $q_{fin} = 3$ would lead to infinitely many words to the language recognised by the automaton.

Control on q_{fin} and Add path

If there exists a word in $\{x_0, \ldots, x_{k-1}\}$ that is a prefix of x_k then

- if $p \neq q_{fin}$ we add p to the set of final states and the construction is terminated.
- if $p = q_{fin}$ then we do a transformation as in the example.

In all cases we add a path from p to q with label w.

Control on q_{fin} and Add path

If there exists a word in $\{x_0, \ldots, x_{k-1}\}$ that is a prefix of x_k then

- if $p \neq q_{fin}$ we add p to the set of final states and the construction is terminated.
- if $p = q_{fin}$ then we do a transformation as in the example.

In all cases we add a path from p to q with label w.

Main result

Theorem

For each $k \in \{0, \ldots, m\}$, the language recognised by the automaton $\mathcal{A}_X^k$ is $L(\mathcal{A}_X^k) = \{x_0, \ldots, x_k\}$.

Construction algorithm

Theorem

Let $X = (x_0, \ldots, x_m)$ be a list of words in A^* ordered by right-to-left lexicographic order and let $\sum_{i=0,m} |x_i| = n$. There is an algorithm for the construction of the automaton $\mathcal{A}_X^m$ recognising X in $\mathcal{O}(n)$.

Construction-$\mathcal{A}_X$(X)

1. $(\mathcal{A}, R) \leftarrow$ Construction-$\mathcal{A}_X^0$ $(X[0])$
2. **for** $k \leftarrow 1$ **to** $|X| - 1$ **do**
3. $\quad (\mathcal{A}, R) \leftarrow$ Add-word$(\mathcal{A}, X[k], X[k-1], R)$
4. Return $\mathcal{A}$

Non minimality of the automaton

Example

$X = (\mathtt{aaa}, \mathtt{ba}, \mathtt{aab}, \mathtt{bb})$

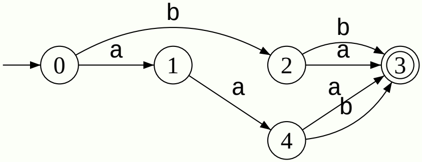


Figure: $\mathcal{A}_X^3$

$\mathcal{A}_X^3$ is not minimal since the states 2 and 4 are equivalent.

Non minimality of the automaton

Example

$X = (\texttt{aaa}, \texttt{ba}, \texttt{aab}, \texttt{bb})$

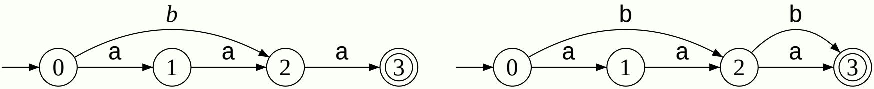


Figure: $\mathcal{A}_X^1$, the incorrect construction of $\mathcal{A}_X^2$

In this example *bb* is also in X.
In this case the indegree control is not necessary.

Set of suffixes of a given word

- Let y in A^* and $S(y)$ be the set of suffixes of y.
- $S(y)$ sorted by decreasing order on the lengths of the elements in $S(y)$.
- denote by $\mathcal{A}_y$ the automaton $\mathcal{A}_{S(y)}$ and by $\mathcal{M}_y$ the minimal automaton of $S(y)$.
- $\mathcal{A} \longrightarrow \sharp\mathcal{A}$ the number of states of $\mathcal{A}$.

We consider the ratio $D(y) = \frac{\sharp\mathcal{A}_y}{\sharp\mathcal{M}_y}$.

Set of suffixes of a given word

We have done experiments on the set of suffixes of a given word.

- $D_n^{max} \longrightarrow$ the greatest of $D(y)$ with y of length n.

n	D_n^{max}
10	1.83
15	2.41
20	3.04

- $D_n^{max} \leq 4$ for words y with $|y| \leq 20$.
- Bad cases linked with words powers of a short one with great exponent

Set of suffixes of a given word

- $D_n \longrightarrow$ the greatest ratio among the $D(y)$
- In each column we have $\mathcal{D}_n$ for a set of generated words which either are not powers of the same word or are powers of a word with an exponent less than a fixed number.

n	$exp < 3$	$exp < 2$	$exp < 1$
10	1.75	1.66	1.54
20	2.22	2.16	2.42
30	2.16	2.22	2.24
50	1.96	1.85	2.60
100	1.60	1.71	1.79

The experimental results are good in general even if they do not show clearly our conjecture.

Set of suffixes of a given word

- $D_n \longrightarrow$ the greatest ratio among the $D(y)$
- In each column we have $\mathcal{D}_n$ for a set of generated words which either are not powers of the same word or are powers of a word with an exponent less than a fixed number.

n	$exp < 3$	$exp < 2$	$exp < 1$
10	1.75	1.66	1.54
20	2.22	2.16	2.42
30	2.16	2.22	2.24
50	1.96	1.85	2.60
100	1.60	1.71	1.79

The experimental results are good in general even if they do not show clearly our conjecture.

Set of suffixes of a given word: modified construction

PF control and Height control are not necessary in this case.

Lemma

Let y in A^* and y_k in $S(y)$ such that y_k is not a prefix of a word in $\{y_0, \ldots, y_{k-1}\}$. Then we have that $PF(q) = 1$.

Lemma

Let y in A^* and y_k in $S(y)$ such that y_k is not a prefix of a word in $\{y_0, \ldots, y_{k-1}\}$. Then we have that $H(p) > H(q)$.

Set of suffixes of a given word: modified construction

- We propose a modified Indegree control in order to avoid equivalent states as in the example.
- We expect that an improved version of the algorithm actually builds the (minimal) suffix automaton of y.

Open problems

- Find a general upper bound for ratios D
- Does there exist an on-line construction for the minimal automaton accepting a finite set of words that runs in linear time on each word being inserted in the automaton?

Open problems

- Find a general upper bound for ratios D
- Does there exist an on-line construction for the minimal automaton accepting a finite set of words that runs in linear time on each word being inserted in the automaton?